WHAT IS CLAIMED IS:

1. A method for enforcing static and dynamic access policy protecting a resource in a computer system, the system having a client thereof making a first access request for the resource, the method comprising:

determining a static maximum allowed access data structure pursuant to an evaluation of the first access request, wherein the static maximum allowed access data structure includes information representative of a set of policies that is reduced to static form that is common to a class of access requests;

storing the static maximum allowed access data structure; and

in response to a determination that the static maximum allowed access data structure is applicable to a second access request, utilizing said static maximum allowed access data in connection with the requested permission set of the second access request.

2. A method according to claim 1, wherein the storing of the static maximum allowed access data structure includes storing the static maximum allowed access data structure in cache memory.

3. A method according to claim 1, further comprising computing a client security context after the first access request for the resource is received from the client.

4. A method according to claim 1, further comprising determining whether said second access request is granted based at least in part on dynamic data and dynamic policy algorithms.

5. A method according to claim 1, further comprising:

evaluating whether the requested permission set of the second access request is represented within the static maximum allowed access data structure.

6. A method according to claim 1, wherein an application programming interface utilizes said static maximum allowed access data structure in connection with the evaluation of the requested permission set of the second access request.

7. A method according to claim 1, further comprising:

evaluating whether the requested permission set of the second access request is contained within said static maximum allowed access data structure.

8. A method according to claim 1, further comprising:

evaluating whether there is at least one dynamic access control entry in a discretionary access control list associated with the second access request.

9. A method according to claim 1, further comprising:

evaluating whether at least one access control entry in a discretionary access control list associated with the second access request is a deny access control entry.

10. A method according to claim 9, wherein if there is not at least one deny access control entry, the method further comprises:

evaluating whether the requested permission set of the second access request is encompassed by (1) permissions obtained by evaluating at least one dynamic grant access control entry and (2) permissions contained said static maximum allowed access data structure.

11. A method according to claim 1, wherein said determining of a static maximum allowed access data structure pursuant to an evaluation of the first access request supplements a standard determination of access rights based upon static and dynamic policy data and policy evaluation algorithms.

12. A computer readable medium having computer executable instructions for carrying out the method of claim 1.

13. A modulated data signal carrying computer executable instructions for performing the method of claim 1.

14. A computer readable medium bearing computer executable instruction for carrying out a static maximum allowed access mechanism for an application in a computer system having a resource manager that manages and controls access to a resource, wherein the static maximum allowed access mechanism provides extensible support for application-defined business rules via a set of APIs and DACLs, wherein a static maximum allowed access data structure is determined pursuant to an evaluation of a first access request; and wherein data of said static maximum allowed access data structure is utilized in connection with a second access request.

15. A computer readable medium bearing computer executable instruction for carrying out a static maximum allowed access mechanism according to claim 14, wherein the storing of the static maximum allowed access data structure includes storing the static maximum allowed access data structure in cache memory.

16. A computer readable medium bearing computer executable instruction for carrying out a static maximum allowed access mechanism according to claim 14, wherein
a client security context is computed after the first access request for the resource is received from the client and the determination as to whether said access request is granted is dynamic.

17. A computer readable medium bearing computer executable instruction for carrying out a static maximum allowed access mechanism according to claim 14, wherein the utilization of data of the static maximum allowed access data structure in connection with a requested permission

set of a second access request includes an evaluation of whether the requested permission set of the second access request is represented within the static maximum allowed access data structure.

18. A static maximum allowed access data structure stored on a computer readable medium for use in connection with access check determinations for an application in a computer system, the data structure comprising:

an identifier identifying the data structure as a static maximum allowed access data structure; and

data representing the static maximum allowed access for a given security descriptor and a corresponding client context in connection with an access request.

19. A data structure according to claim 18, wherein the static maximum allowed access data structure is stored in cache memory.

20. A data structure according to claim 18, wherein the static maximum allowed access data structure includes a security descriptor.

21. A data structure according to claim 18, wherein the static maximum allowed access data structure includes one of data representing the client context and a pointer to the client context.